Real Time Emotion Recognition through Facial Expressions for Desktop Devices

P. M. Chavan, Manan C .Jadhav, Jinal B. Mashruwala, Aditi K. Nehete, Pooja A. Panjari

Abstract—*Thepaper states the technique of recognizing emotion using facial expressions is a central element in human interactions. By creating machines that can detect and understand emotion, we can enhance the human computer interaction. In this paper, we discuss a framework for the classification of emotional states, based on still images of the face and the implementation details of a real-time facial feature extraction and emotion recognition application are discussed. The application automatically detects frontal faces from the captured image and codes them with respect to 7 dimensions in real time: neutral, anger, disgust, fear, joy, sadness, surprise. Most interestingly the outputs of the classifier change smoothly as a function of time, providing a possibly worth representation of code facial expression dynamics in a fully automatic and unnoticeable manner. The main objective of the paper is the real-time implementation of a facial emotion recognition system. The system has been deployed on a Microsoft's Windows desktop.*

Keywords—Real time, facial expression, emotion recognition.

I. INTRODUCTION

According to Dr. Charles Darwin, the facial expression indeed contribute in communicating one's emotions, opinions as well as intentions to each other in a effective way. In addition to this, his study on human behavior explicitly states that such expressions also provide information about the cognitive state of a person. This includes the states like- boredom, stress, interest, confusion so on so forth. Hence, on the similar lines, considering the importance and the escalating need for advanced Human Computer Interaction, this paper revolves around generating the mood or emotion of humans based on their facial expression. Furthermore, it explains the necessity of Real-Time Systems in order to achieve high levels of interactions with the machines. It is of prime importance that the interaction of humans with the computers should be latency free, thus taking it to the level of face-to-face communication.

Manuscript Received on May 2013.

Prof. P. M. Chavan, Professor, Department of Computer Technology and Information Technology, Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019, India

Mr. Manan C. Jadhav, Student, Department of Computer Technology and Information Technology, Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019, India

Ms. Aditi K. Nehete, Student, Department of Computer Technology and Information Technology, Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019, India

Ms. Jinal B. Mashruwala, Student, Department of Computer Technology and Information Technology, Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019, India

Ms. Pooja A. Panjari, Student, Department of Computer Technology and Information Technology, Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019, India

Hence, we hereby propose the development of a robust real-time perceptive system, which will take into account the facial expressions, detect human face and code the expression dynamics. Thus, systems like working on these lines have a wide range of applications in basic and applied research areas, including man-machine communication, security, law enforcement, psychiatry, education, and telecommunications. As a result, attempts are being made to elevate the levels of usability and ease of interaction with minimum latency (fast response time) as well as minimum error co-efficient in the output In an attempt to make the system more efficient, we suggest an application, wherein, 7 dimensions of facial expressions can be mapped, high level of accuracy can be obtained, and last but the not the least, processing time can be saved.

II. SYSTEM ARCHITECTURE

THERE ARE 10 BASIC BUILDING BLOCKS OF THE EMOTION RECOGNITION AS ENLISTED BELOW:

Live Streaming: Image acquisition

Skin Color Segmentation: Discriminates between face and non-face parts

Face Detection: Locates position of face in the given image frame using skin pixels

Eye Detection: Identifies position of eyes in the image frame

Lip Detection: Determnies lip coordinates on face

Longest Binary Pattern: Skin and non-skinpixels are converted to white and black pixels, respectively

Bezier Curve Algorithm: Applies bezier curve equation on the facial feature points

Emotion Detection: Emotion is detected by pattern matching using valuesfrom database

Database: Stores values derived from bezier curve equation that are used for comparison in emotion detection phase

Output Display: Renders ouptut from detectionphase on screen

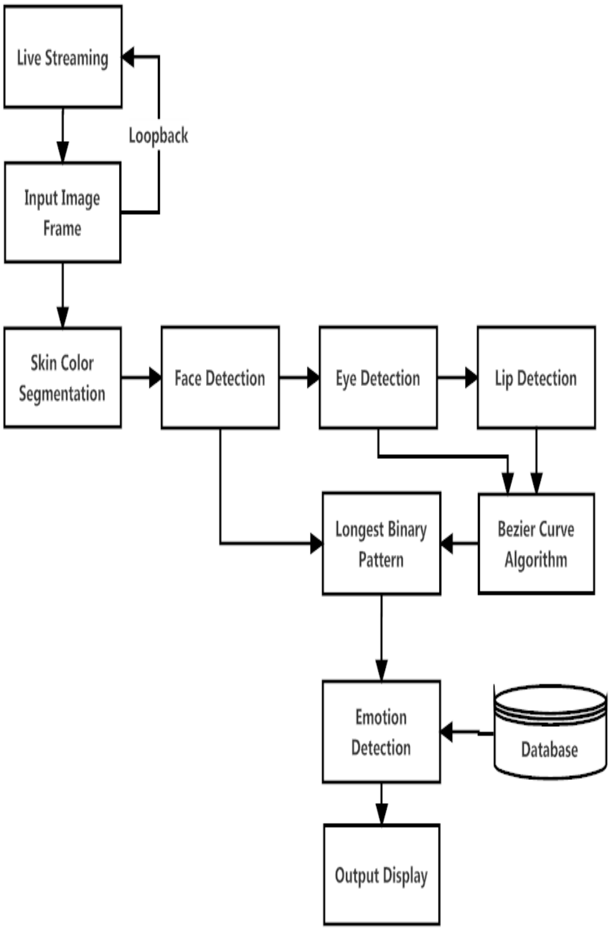

Figure 1. System Architecture

i. Live Streaming

Live streaming is the fundamental step of image acquisition in real time, where image frames are received using streaming media. In this stage, the application receives images from built-in webcam or external video camera device. The streaming continues until input image frame is acquired.

ii. Skin Color Segmentation

Skin color segmentation, discriminates between skin and non-skin pixels. Skin segmentation permits face detection to focus on reduced skin areas in preference to the entire image frame. Skin segmentation has proved to be instrumental as regions containing skin can be located fast by exploring pixels having skin color. Before performing skin color segmentation, we first contrast the image which involves separation of brightest and darkest image areas. Further, the largest continuous region is spotted to identify potential candidate faces. Then the probability of this largest connected area becoming a face is determined and is further used in subsequent steps for emotion detection

iii. Face Detection

Face detection can be considered as a specific case of Object-class detection that discovers size and location of objects, listed in class, within an input image.In this step, face is detected by identifying facial features while ignoring all non-face elements.

To perform face detection image is first converted into a binary form and scan the image for forehead. We then search for maximum width of continuous white pixels till we reach eye brows.We now cut the face in such a way that its height is 1.5 times that of its width.

iv. Eye Detection

For eyes detection too, we convert RGB image to binary image and then scan image from W/4 (where W is width) to remaining width in order to find the middle position of the eyes.Next, we determine upper position of the two eyebrows. For left eye search ranges from W/8 to mid (middle position between eyes) of the imageand for right eye from mid of the image to W-W/8 of the total width. We then place continuous black pixels vertically in order to get eyes and eyebrows connected. To achieve this for left eye, pixel lines are placed vertically between mid/2 to mid/4 and in case of right eye vertical black pixel lines are placed between {mid + (W-mid)/4 } to {mid + 3*(W – mid)/4}. We further search black pixel lines, horizontally, from the mid position to the starting position so as to locate the right side of left eye. To detect left side of right eye we scan the starting to mid position of black pixels within the upper and lower position of right eye. The left side of left eye is simply the starting width of image and right eye's right side is the ending width. Using this we cut the left side, the right side and the upper and lower positions of each eye from the previous RGB image.

v. Lip Detection

Another important feature to be detected to analyze emotions is Lip. It helps us to know whether shape of lips is : plain, slightly curved, broadly curved , pout , parted lips. This module outputs a lip box having dimensions that can contain entire lip in the box and some part of nose.

vi. Longest Binary Pattern

It is extremely important that the area of interest be made smooth so as to detect the exact emotions. So we convert the skin pixels in that box to white pixels and rest of them to black. Then to find a particular region , say lip, we find the biggest connected region of monotone color , here its longest sequences of black pixels. This results in longest binary pattern.

vii. Bezier Curve Algorithm

Most notable applications of Bezier Curve include interpolation, approximantion, curve fitting, and object representation. The aim of the algorithm is to find points in the middle of 2 nearby points and to repeat this until we have no more iterations. The new values of points results in curve. The famous Beizer equation is accurate formulation of this idea.

Formula is :

$$\gamma(t) = \Sigma^{n}_{i=0} P_i\, [n!/(i!*(n-1)!)](1-t)^{n-i} * t^{i}$$

Cubic Bezier:

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

$$B(t) = (1-t)^3p_1 + 3(1-t)^2tp_2 + 3(1-t)t^2p_3 + t^3p_4$$

viii. Emotion Detection

Next step after detecting features is to recognize the emotions. For this we convert all the bezier curve into larger regions. This static curves are then compared with the values already present in database. Then the nearest matching pattern is picked and related emotion is given as output.If the input doesnt match the ones in database, then average height for each emotion in database is computed & on basis of this a decision is made.

ix. Database

Database contains existing values that can be later used in comparisions to find out the closest matching emotion to be displayed as output.

x. Output Display

Once the comparision is done, the output will contain the graphical form as well as an animated figure describing the emotion of the portrayed input.

III. KNOWLEDGE BASE

In our database, there are two relations. First relation "Input" contains name of the poser and their index of 7 kinds of emotion which are stored in other relation "Orientation".
In the “Orientation” relation, for each index, there are 6 control points for lip Bezier curve, 6 control points for left & right eye Bezier curve each, lip height and width, left eye & right eye height and width . Applying this approach the system understands and maps the emotion of the captured input.

IV. SKIN COLOR SEGMENTATION

Human face comprises of varied features namely eyes, nose, lips, forehead, hair, eyebrows etc. For detecting emotions, some features like eyes, lips et al are required while some others like hair, ear et al are not. We need to sort the face into fragments which will separate face and non-face elements. This is achieved by using the image processing technique called Image Segmentation. This generates segments of adjacent pixels which share similar visual characteristics and thus makes it easier to analyse the facial features.

These segments are thus mapped and largest connected region which corresponds to similar characteristics like intensity or RGB value are taken. If this region has dimension ratio (height: width ratio) between 1 and 2, then it can be speculated that the region has to be a face. Explicit detection of facial features is done hereafter.

V. FACE DETECTION

Once segmentation is done, it becomes easier to analyze pixels having similar and different visual characteristics. The emotions depend on shape of eyebrows, lip curves, eye movement et al. For this particular region, explicit detection of such features is required. So prior to their detection, we should recognize a face frame so that we can generate the dimensions and work with the corresponding dimensions for further mappings.

Now, for detection of such a frame, the first step is to convert RGB image into a binary image containing a series of black and white pixels. Hence, the RGB values of all the pixels are taken. An average value of RGB for each pixel is calculated and if it is below 110, then the corresponding pixel is substituted by a black pixels. On the other hand, if it is greater than 110, it is substituted by white pixel. Thus a binary image is generated.

Now, we start scanning the binary image from the midpoint of the edge of the image. Since we are bound to come across hair first, we get a series of continuous black pixels. The moment forehead appears, it changes to white pixels. This point is marked. Now, vertical scanning from middle axis of the image in both sides-left and right is done to find the maximum width of our face frame. When we come across eyebrows while scanning, we find black pixels. From middle line, vertical scan will have no eyebrow portion, hence, only white pixels will be encountered. We keep scanning in both sides- left and right until the width drops to half of the previously mapped width. This will mark the end of eyebrow portion. Once we get the final width, we calculate the length of frame as 1.5 times of the width. From the starting point of forehead which was marked earlier, the length of frame is measured and a rectangular frame is created. This is the face region we need to work on from a full-fledged image.

VI. EYE DETECTION

We now have the rectangular frame with width (w) and length ($l=w*1.5$). We know that our face exhibits symmetry. Hence, finding a middle (central) axis is of utmost importance. We cannot directly map the axis at $w/2$. This is because we need to avoid any type of discrepancy in our detection. Hence, we scan the frame from $w/4$ to $w-w/4$. As mentioned in Face Detection Module, the central part of face gives a series of continuous white pixels. Hence, whenever (around $w/2$) such series is obtained, we mark it as central axis (*mid*).

We proceed by finding the upper starting point of both the eyebrows. This is done by finding black pixels while scanning vertically from $w/8$ to *mid* for the left eyebrow and *mid* to $w-w/8$ for the right eyebrow.

Eyes and eyebrows, when combined together will point to exact eye movement. Hence, while detecting an eye, the eyebrow and eye should be kept connected. This is done by placing black pixels in the white places between an eye and an eyebrow. But, since for different faces, this lengths/space may differ, we pick a standard function of width of the frame and central axis. As a result, for the left eye, we place the black pixels from *mid*/4to *mid* /2 and for the right eye, we place the black pixels from $mid+(w-mid)/4$ to $mid+3*(w-mid)/4$. The height for the placing of pixels is calculated to be from position of eyebrow to (h-position of eyebrow)/4.h stands for the height of the image. Next, we find the lower positions of both the eyes. Again, this is done by searching black pixels vertically. Scanning from mid/4 to mid-mid/4 for left eye and symmetrically scanning from lower end of image to starting point of eyebrow from mid+(w-

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

mid)/4 to mid+3*(w-mid)/4 for the right eye.

Thereafter, the end points: left side of right eye and right side of left eye are plotted. This is achieved by searching black pixels horizontally from middle point to starting point of black pixels between the upper and lower positions of left eye, and from middle point to starting point of black pixels between upper and lower positions of right eye.

Thus, we get a rectangular frame comprising of eye and eyebrow feature. The frame starts at left side of left eye and ends at right side of right eye. Corresponding area is cut from the RGB image. In this way, the eyes are detected in a formulated manner, which does not hinder various dimensional issues for different types of faces.

VII. LIP DETECTION

Just as eye detection is done, similar type of frame is to be carved out for the lips. The shape of lips: plain, slightly curved, broadly curved, pout, parted lips (revealing teeth); is one of the important features for detecting emotion. Happiness, Anger, Excitement, Sadness, Concealment, and Nervousness et al can be judged if the movement of lips is known.

Thus, we need to determine a lip box having such dimensions that the entire lip be contained in the box itself. Again, this must be purely dimensional, custom made for each of the different faces and hence, it becomes a function of distance between different features in all.

Firstly, we determine the distance between forehead and eyes. We then add this distance to the lower point of eye. These dimensions are based on the symmetry that our face represents. Now, this final sum gives us the upper height of lip box. The ending height (lower height) of this box will be the lower end of the face. Moreover, the width is calculated as distance between the ¼ position of left eye box and ¾ position of right eye box.

The rectangular lip box thus obtained contains entire lip and may contain some part of nose.

VIII. APPLY BEZIER CURVE ON LIP

The rectangular lip box contains lip, which is inarguably the most varied feature of our face. It may have variations in tone and evenness. It is of utmost importance that the lip area be made sufficiently smooth so as to detect exact movements of lips and the trenches it makes in the face.

We convert the skin pixels in the lip box to white pixels and rest of them to black pixels. The tone of skin, as we know changes unevenly in this area (upper lip, chin, dents in cheek etc).Hence, these pixels which are similar to skin pixels are also converted to white pixels. If average RGB value of pixels is calculated, and the difference between two such pixels is less than or equal to 10, then we consider them as similar pixels.

While finding the similar pixels, 10 can not always be the standard comparison value. Depending on the quality of the image, we will get variations in the RGB values. To tackle this issue, we first use a histogram for finding the lowest and the highest RGB value. If this difference is less than 70, then the quality of corresponding image is high. Hence, to judge similar pixels, we will need to consider 7 as a standard for comparison instead of 10. However, if the histogram gives the difference between lowest and highest RGB value greater than 70, then due to the low quality of the image, we will need to consider 10 as the comparison factor.

To detect just the lip, we find the biggest connected region. This can be cross checked using the fact that in out rectangular lip box, lip forms the largest area which comprises of mono-tone and which is different than the skin.

Thereafter, we apply Bezier curve on the black pixilated lip region that is obtained from above steps. In order to apply it, we find start and end point of lip in horizontal direction by scanning for continuous black pixels and stopping when a white pixel is encountered.

We proceed by drawing tangents from each end pixel of the lip for each of the upper and lower lip. We then find 2 points on the tangents which do not lie on the lip. We then slowly format the tangents and the 4 (2 from upper lip tangent and 2 from lower lip tangent) points on it, to give an outline curve of the lip area. Explicit use of Cubic Bezier Curves is done in this process.

For instance, if the points are p_1, p_2, p_3, p_4, then explicit form of curve is given by:

$$B(t) = (1-t)^3 p_1 + 3(1-t)^2 t p_2 + 3(1-t)t^2 p_3 + t^3 p_4$$

From these 4 distinct points, we can create a cubic beizer curve that goes through all 4 points in order and gives us the outline curve for mapping our values in the movement of lips.

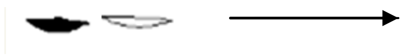

Figure 2. Bezier Curve on Lips.

IX. APPLY BEZIER CURVE ON EYES

Emotions like Anger or Sadness can be directly judged based on the size of the eye. If the eyes are wide then, it might point to surprise or anger, while deep eyes may point to sadness. Hence, an outline curve of eyes is also needed to be computed.

While applying Bezier curve on the eyes, we need to remove the eyebrows from the eye. For doing so, we first search continuous black pixels and then continuous white pixels followed again by continuous black pixels in the binary image of the detected eyes. By just removing the first series of black pixels, we eliminate the eyebrows.

Similar to what we did while applying Bezier curve to lips, we find the largest connected region to segregate eyes and eliminate skin and skin-alike pixels owing to mono-tone effect. Once again , we mark the end pixels and get four points on the tangents so that for each eye, we can represent the area by a cubic Bezier curve.

Figure 3. Bezier Curve on Eye.

X. EMOTION DETECTION

Once all the feature detection is done, all that remains is comparison. We convert all the Bezier curves (left eye, right eye, lips) into larger areas with width 100 units and height according to its width (h=1.5w). These static curves are then compared with the existing values in the database. Depending on the comparison, nearest matching pattern is picked and the corresponding emotion is given as output.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

If the input doesn't match the ones in the database, then average height for each emotion in the database is calculated, and decision is generated on the basis of average height. The closest matching emotion is then given as output.

The output after comparison with the database would be of the graphical form as well as an animated figure depicting the emotion as it is!

Figure 4. Facial Feature Detection and Emotion Recognition

XI. CONCLUSION AND FUTURE WORK

In this paper we have proposed a system that automatically detects human emotions on the basis of facial expressions. This paper and the techniques followed are more suited as this paper already uses the best of other approaches for face recognition, thus limiting the latency in response. Moreover, it further takes a step to improvise the emotion detection technique using Cubic Bezier Curve Implementation which is more adaptive and resurface as the ones with utmost importance in various other fields like that of robotics, computer graphics, automation and animation. The system works well for faces with different shapes, complexions as well as skin tones and senses basic six emotional expressions.

The system even handles face rotations across x-axis and fetches accurate results for horizontal rotations. However this flexibility is not extended over vertical rotations of image.

Although it is unable to discover compound or mixed emotions.The system accurately detects the emotions for a single face, however this correctness reduces prominently for multiple faces. The system can further be upgraded to include multiple face detections.

This would indeed form a big step in areas of Human Computer Interaction, Image Processing, Animation, Automation and Robotics.

REFERENCES

1. MarianStewart Bartlett, Gwen Littlewort, Ian Fasel ,Javier R. Movellan, Real Time Face Detection and Facial Expression Recognition: Development andApplications to Human Computer Interaction,Machine perception laboratory Institute for Neural Computation, University of California, San Diego, CA.
2. Liyanage C De Silva, Suen Chun Hui, Real Time Facial Feature Extraction and Emotion Recognition 2003.
3. Abdesselam Bouzerdoum, Son Lam Phung, Fok Hing Chi Tivive, Peiyao LiFeature, Selection For Facial Expression Recognition .
4. Irfan A. Essa,Coding, Analysis, Interpretation, and Recognition of Facial Expressions.

AUTHORS PROFILE

Prof. P. M. Chavan, Professor (Computer Technology and Information Technology Department) at Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019

Mr. Manan C. Jadhav , Student at Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019

Ms. Aditi K. Nehete, Student at Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019

Ms. Jinal B. Mashruwala, Student at Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019

Ms. Pooja A. Panjari, Student at Veermata Jijabai Technological Institute, Matunga, Mumbai- 400 019

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication